(6)

# Analysis of Naturally Occurring Slip-Stick Data in Arctic Ice Floes


Peter Michaux
Institute of Ocean Sciences
9860 West Saanich Road
P.O. Box 6000
Sidney, British Columbia, Canada
V8L 4B2




Physics Co-op Work Term Report

in partial fulfillment
of the requirements of the Physics Co-op Program

Summer 1996


Peter Michaux

Department of Physics and Astronomy
University of Victoria

## Abstract

Naturally occuring slip-stick data was recorded from a closing lead in the arctic in 1994. A portion of that data is analysed here with the idea of a slip-stick stress release model in mind.

# Table of Contents

## List of Figures

## Introduction

When wind and ocean currents interact with sea ice at the air-ice and water-ice interfaces they push on the ice. This force can cause internal stress to build in the comparatively immobile ice. When the magnitude of the stress exceeds the strength of a weak support in the ice, the stress is released through the failing support. There are several possible modes of stress release; for example, the ice could crack. Here we are interested in release by the slip-stick process.

One very well known example of the slip-stick process is the old creaky door hinge. When a door with such a hinge is opened slowly the part of the hinge attached to the door slides over the other, attached to the wall, discontinuously. In fact, at low speeds, the hinge slips and when enough stress is released it sticks. When the stress grows greater than the static friction limit of the interface, the hinge slips again. During the stick the door is silent and as it slips acoustic energy, along with heat, is released. The intermittent nature of this sound source can be heard with the ear if the door is opened slowly enough.

This process can take place when one ice sheet is rubbing over another as the two are pushed together or pulled apart by wind and water currents. Acoustic waves are released into the water and seismic waves into the ice. By listing to the slip-stick sounds generated, it is hoped that the stress released and the overall internal stress of the ice can be determined. This has many possible practical applications. One of the most important is for ship travel. It is faster, more cost effective, and safer for an ice breaking ship to travel through low stress ice. If a grid of hydrophones could be placed on major ship routes to listen to the rubbing of adjacent ice sheets, the ice under lowest stress could then determine the ship's course.

To achieve this, we need a slip-stick model for sea ice interaction. Before a model can be produced the characteristic parameters of the process need to be determined from artificial and naturally generated data. Describing the parameters of slip-stick process with the idea of a stress release model in mind is discussed.

## Data Collection and Storage

In April 1994, about seventy nautical miles north of Prudhoe Bay, an open water lead[1] was found by air. A new thin (about two to five centimetres thick) layer of ice had formed over the lead and had subsequently cracked. The lead was closing and the two thin ice floes were interacting by the slip-stick process. Two hydrophones were placed five meters under the ice to record the acoustic and seismic waves. They were about forty

[1]A lead forms when an ice floe cracks and pulls apart exposing the ocean beneath it.

meters apart on a line perpendicular to the lead. On April 16 and 19, a total of about five hours of data was recorded.

Data was collected at 44.1 kHz per hydrophone. It was interleaved and stored as a single stream on VCR tape (one tape for each day). To distinguish the two hydrophones, the least significant bit of each integer from hydrophone one, nearest the lead, was set to zero making the data all even. The data from hydrophone two was made odd. The data was recorded as integers in arbitrary pressure units. Multiplying these integers by 1.152e-3 Pa converted them to numbers with pressure units of Pascals (Pa).

Because no time annotations were made on the storage tape absolute times of the various signals are not known. This is not a serious hindrance as relative times of signals are important. When a portion of the data is logged onto computer, the file is named after the date it was recorded and the VCR count at which it began. The start of the tape is set to zero VCR counts. For example a file containing data that was collected on the sixteenth at VCR count 1334 is called l6r1334.pcm. Along with the time (in seconds) from the start of the file, any part of the data can be named.

## Raw Time Series Description

The first step of the analysis was to listen and look at the raw hydrophone time series to find groups with common characteristics. While listening to the data from the sixteenth it became apparent that three main sound classes existed. They were called: singles, triples, and continuous rubbing. One feature common to all three classes was their length. Most of the signals that were loud enough to be recognized clearly out of the noise only lasted up to about twenty seconds with a few lasting up to about one minute.

A raw time series sample of singles (from l6r0928.pcm) is shown in figure 1. Singles sound much like an automatic machine gun firing continuously. The individual slips are visible beginning very abruptly at 2612, 2787 and 2958 ms. This is one of the best examples of the singles and the signal to noise ratio is only about 1.5:1.

Triples sound like a cycle of three distinct slips. The signal to noise ratio and the signal structure for the samples of triples makes it difficult to look at the raw time series even though it is recognisable to the ear. It is easier to see the structure by looking in the frequency domain. Figure 2 shows five seconds of triples data from l6r2120.pcm. A complete cycle takes place between 1800 and 1400 ms. The first and most prominent sound comes at 1800 ms. Although only two sounds can be distinguished by ear following this first one, several or a continuous sound appears on the plot between 1200 and 1400 ms. Perhaps these following sounds are analogous to earthquake aftershocks. Neither

singles nor triples received much analysis attention as the continuous signals were so intriguing.

Continuous rubbing is the case where the individual slips cannot be resolved by ear alone. The relatively long duration of the samples and the good signal to noise ratio for the continuous rubbing made it easier to analyse and gather good statistics than the other categories. However, the structure of the signals in this class is what made them the focus of attention. The samples l6r1280.pcm and l9r4291.pcm were investigated in the most detail.

Figure 3 shows one signal from the l6r1280.pcm sample. The signal has been received in three stages. The first wave to arrive is the P-wave at about 219.126 s and is about 7 ms. The SH-wave follows it arriving at 219.143 with a duration of about 10 ms. Both the P and SH-waves travel through the ice and any initial high frequency components are attenuated leaving observable only the long period waves. The last arrival is the acoustic wave at 219.153 s. The acoustic wave travels through the water and so its high frequency content is retained. It is difficult to determine the length of the acoustic wave because the noise also contains high frequency. The separation of the three waves indicates that the sound source is relatively far away as the three waves travel at different speeds.

In figure 4 one can almost immediately see a relationship between the height of each SH-wave and the time between it and the preceding signal. We will call this time the period of the signal. The signals are very large around 219.1 s and have long periods. However around 219.4 both the signal height and period have decreased. For a short 15 s, section of data including that in figure 4 the height versus period has been plotted in figure 5. This figure indicates there is a linear relationship between height and period. A further discussion of this plot is given in the discussion.

Figure 6 shows a sample of the l9r4291.pcm data. This sample sounds and looks quite different from l6r1280. Although the individual pulses cannot be heard in either, 19r4291 is a much higher and more squealing sound. We cannot see distinct wave arrivals which may imply that the sample is from the near field and all the different wave modes are super imposed. Individual adjacent pulses are not similar in most cases and the envelope of the time series changes sporadically.

## Signal Processing

### 16r1280.pcm

To extract the parameters of the signal, a method for finding each pulse was required. One or more unique characteristics of the pulse needed to be determined. These

unique characteristics could then be used to locate the signals. These characteristics might include pulse height, shape, or frequency. In this case, height and frequency were used along with the spacing of the SH and acoustic waves. A primary threshold was set at a level that was intended to be broken by the large SH-wave (usually about 1.5 Pa.) To reduce false triggering from unusually large noise the prospective signal then had to satisfy another condition. The signal was high pass filtered (with a cut-off frequency of 2000 Hz) to extract only the acoustic wave. The filtered acoustic wave then had to break another lower threshold (about 0.4 Pa.) However, it had to break the threshold in a specific time window (3-7 ms) after the principle maximum of the SH-wave. The search for the next SH-wave began at a specified length, the window length, of time after the one previously found.

This technique worked quite well for the variability of the signal. The data was processed in five-second segments and the pressure thresholds and time window lengths could be adjusted for each data segment. It was then checked by eye that the routine did, in fact, find almost all of the pulses while falsing as little as possible (the matlab file plots.m was used for this purpose.) If too many were missed or too many false pulses were included the parameters could be adjusted and the process repeated to achieve satisfactory results. The matlab M-file called double.m, named for it's two thresholds, is given in appendix 2.

**l9r4291.pcm**

Several methods were tried for analysis of the l9r4291.pcm data. The first method to find the period involved autocorrelation. A small section of data was chosen at the start of the file that contained about five to ten pulses. It was then autocorrelated. The largest correlation peak is when there is zero lag in the correlation. The next largest peak occurred, in most cases, when the lag was such that one peak was lined up with the following one. This lag was the average period for the signals. The segment window was then moved down the data and the process repeated. An example plot is shown in figure 7. The time series of the period which is also the lag of the second peak, can then be found by finding the second peak. This, however, can be difficult as the second peak becomes small. This can be caused by a quickly changing period or by a section with very different pulse shapes. This method was abandoned for these problems.

The second method investigated used correlation. One pulse that was thought to be representative of majority of pulses was chosen and correlated with the entire file. Correlation peaks were then found by setting a threshold and triggering on the strong correlations. This method resulted in an unsatisfactory number of misses and falses as the pulse shape changed so drastically over time. Rather than using only one pulse an

averaged pulse was used to try to be more representative; unfortunately, this did not improve the results much. Both of these correlation methods were very time consuming and a simpler, faster method was required.

The fastest and most effective method used was very simple. A search for long drops in pressure was performed. When a local maximum was found, the pressure time series was followed to find the subsequent local minimum. If this continuous drop was greater than a certain threshold, the maximum-minimum pair was considered a pulse. Although this may sound too general to find the true pulses, when looking at the data this worked very well and was comparatively fast. Appendix 3 gives the matlab finder.m file used for the search.

A time series plot of period and height was required. Before it could be made the data needed to be despiked. If one pulse was missed by finder.m the period of the following pulse would be about double what it should have been. To get a better picture of the period series these points needed to be removed. To do this despike.m was written. If a period of one pulse deviated from the period of the last point by too much or if it deviated from the low-pass filtered period series by too much the pulse is considered a spike and it is thrown out. Appendix four gives the matlab despike.m file used for despiking the data. The plot in figure 8 was generated by despike.m.

## Discussion of Characteristics

### 16r1280.pcm

Once the signal parameters had been extracted using double.m many plots could be made to look at these parameters in different ways.

Figure 9 shows the conditionally sampled (based on the time of arrival of the acoustic wave) and averaged signal from both channels. If we know that the P-wave travels through the ice at 2300 m/s and the SH-wave at 1680 m/s while the acoustic wave travels at 1500 m/s in the water we can approximate the range of the source. We can do this by using data from only one of the hydrophones. From figure 9 we see that the P-wave arrives at about 14.8 ms, the SH-wave at about 30.9 ms and the acoustic at 40.0 ms. We can calculate the distance using the three combinations of any two waves. We find that the mean source-hydrophone separation is 110 +- 14 m where the standard deviation of the three combinations has been used to estimate the uncertainty.

With a symmetry ambiguity the angular position of the source can be computed using data from both hydrophones. The acoustic wave arrives at hydrophone one at 38 +- 0.5 ms. Using the arrival time of the acoustic wave at hydrophone two and it's now known separation from the source we can find the hydrophone two-source separation. It is

107 +- 14.75 m. These two separations are consistent and we can conclude that the source and two hydrophones approximately form an isosceles triangle with the hydrophones opposite the equal sides.

The P and SH-waves travel along the ice and then through the water to the hydrophone. The different velocities of these waves in water has not been taken into account. To a first approximation this is quite good as the hydrophone depth was much less than the source-hydrophone separation.

If the two source-hydrophone separations are almost equal one might wonder why the heights of the three waves are smaller at hydrophone one than at hydrophone two. It is difficult to say for sure; however, Y. Xie (personal communication) has suggested that the energy produced in acoustic and seismic modes may not be released symmetrically. This could be the result of many different mechanisms.

Two more plots are made automatically after running double. Figure 10 shows the averaged spectrum of the high-passed filtered acoustic signal. The cut-off frequency was 1000 Hz which is well below the frequency peak at about 6440 Hz. The final plot made was the exponentially fitted, conditionally sampled, high-pass filtered, acoustic signal shown in figure 11.

Several other plots were made using the data produced in files by double.m. The plots in figures 12 and 13 were produced using param.m (given in appendix 5.) These plots show the statistics of the parameters from the individual pulses. Some of the terms used to describe the shape of each pulse are: principle height, the difference in height of the SH-wave maximum and minimum; principle max or max, the maximum pressure of the SH-wave; principle min or min, the minimum pressure of the SH-wave; previous max, the local maximum in the SH-wave just before the principle maximum; high frequency height, the difference of the high-pass filtered acoustic wave maximum and minimum; high freq trigger, the time at which the high frequency acoustic wave broke the threshold.

We can see in these plots that the internal parameters of the pulses stay fairly constant over the duration of the signal. In the three plots where principle height is plotted along the horizontal axis, the variance of the other parameter increases with a decrease in principle height. This is most likely due to a poor signal to noise ratio making it more difficult to find the other parameter of interest in the signal. As the mean does not change noticeably for these small signals it is not of great concern.

The plot of the spacing of the principle max and the high frequency trigger shows an increase after the gap in the data at about 235 s. This increase is accompanied by a decrease in the high frequency/principle height ratio. These two changes may indicate that the source moved slightly further away and is hence not as loud.

The histogram plots accompanying the scatter plots also show the mean and standard deviation (std), both with confidence intervals (ci), from the fit of a normal distribution. Also given are the variance (var) and interquartile range (iqr) of the data sample.

Figure 14 shows the time series of the mean height, mean period, and number of counts. Data was taken in 1000 ms windows and these three quantities computed. The window was moved along 50 ms and the process repeated. One can see relationships between these plots. As the period increased so did the height. These reflect the idea that the longer the stress builds, the greater the subsequent release. As the period increases the count rate decreases.

To look at the stress release relationship in more detail a least squares linear fit was performed on the each window of height versus period data. The four plots in figure 15 were produced. The slope of the relationship changes over the duration of the signal. This could be due to changes in external environmental variables like wind speed or direction. The variance of the correlation coefficient jumped drastically at about 232 s. This reflected a poor signal to noise ratio that continued for the following 4 s and many false and missed signals. These four seconds were not analysed for this reason.

Figure 16 shows several unaveraged, unfiltered time series. Variations of frequency peak of the acoustic wave are seen in figure 17. The magnitude increased with increased principle height. The location of the frequency peak stayed fairly constant throughout the data sample.

It is unfortunate that much longer data samples of this type are not present in the recording. Longer samples would likely help understand the slope changes of the height versus period relationship over time. An understanding of these changes may be one of the most important factors in developing a slip-stick model.

**19r4291.pcm**

The period and height time series of this continuous rubbing signal are shown in figures 8 and 18. In figure 8 we can see that not only are the two series out of phase but the phase lag changes. From looking back at the raw time series it was suggested that more than one frequency was dominating the time series. The spectrum shown in figure 19 indeed shows this to be the case. The ice is a wave guide for the P-wave and a fundamental (~750 Hz) and overtones (~1600, 2400, 3200 Hz) are present. The overtones attenuate faster than the fundamental and so they are not as strong. Using the frequencies of the harmonics the source distance can be computed. Further analysis of this signal is required.

## Conclusion

There are still many different areas of this problem to be investigated. This analysis is only a beginning in understanding the naturally occurring slip-stick process in ice. A good understanding of the stress release process is needed. The relationship between pulse height and period will probably be an important part of the understanding. As the slip-stick signals vary so much based on environmental parameters not recorded by the hydrophone, many challenges still need to be overcome before a slip-stick model can be used to determine the stress in ice.

Figure 1: Raw time series sample of l6rm0928.pcm

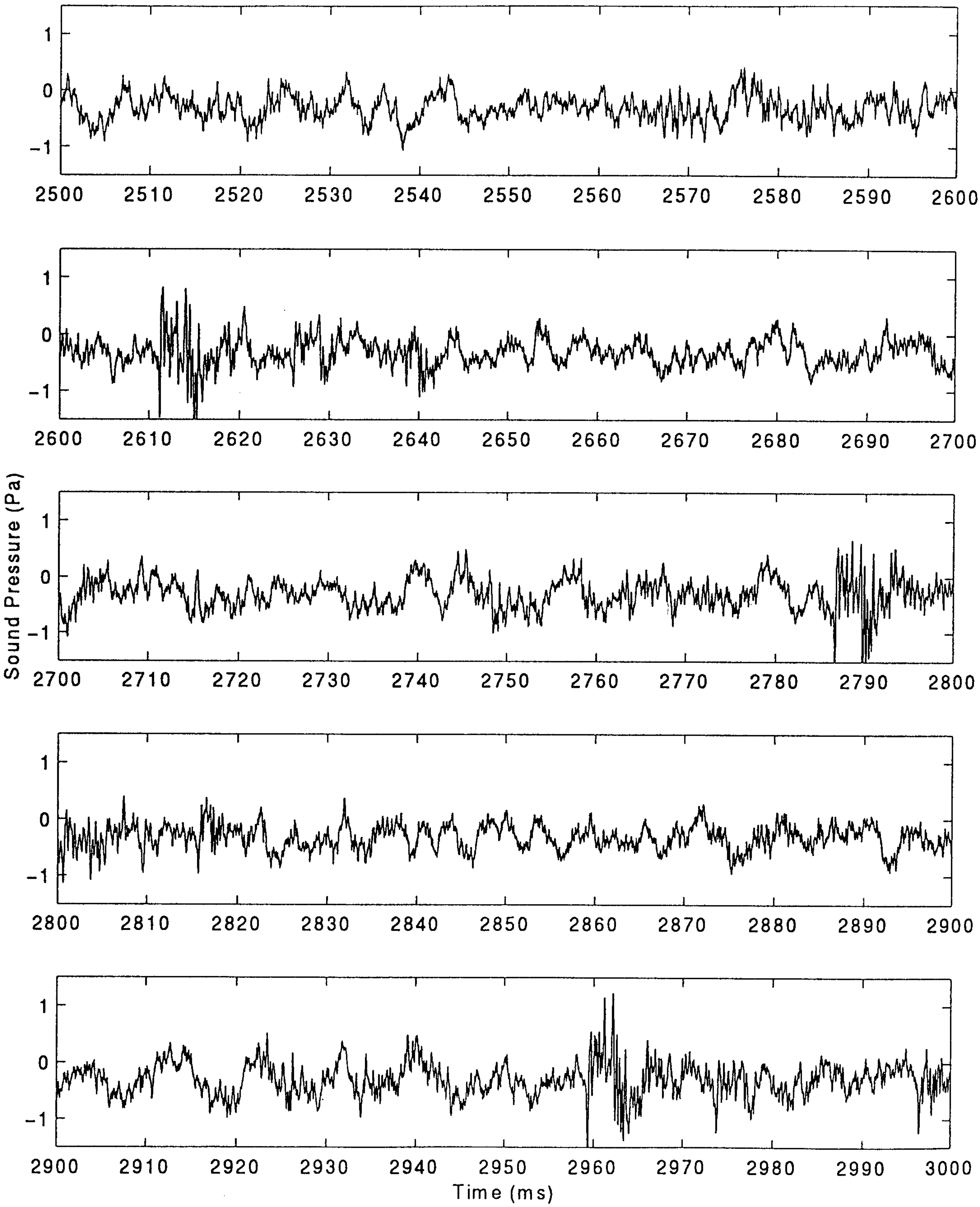


Figure 2: Spectrum of l6r2120.pcm

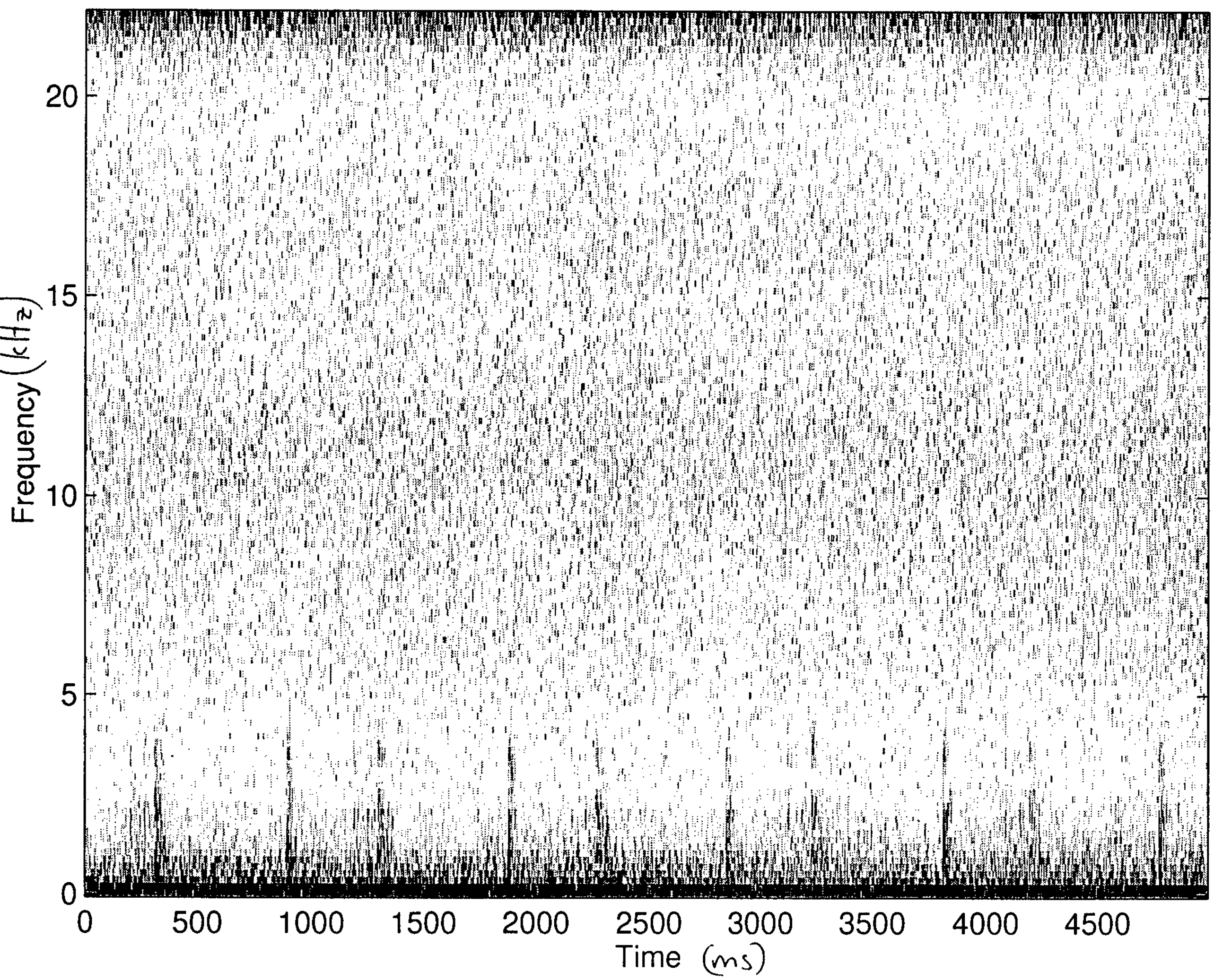


Figure 3: Raw time series sample from l6r1280,pcm small scale

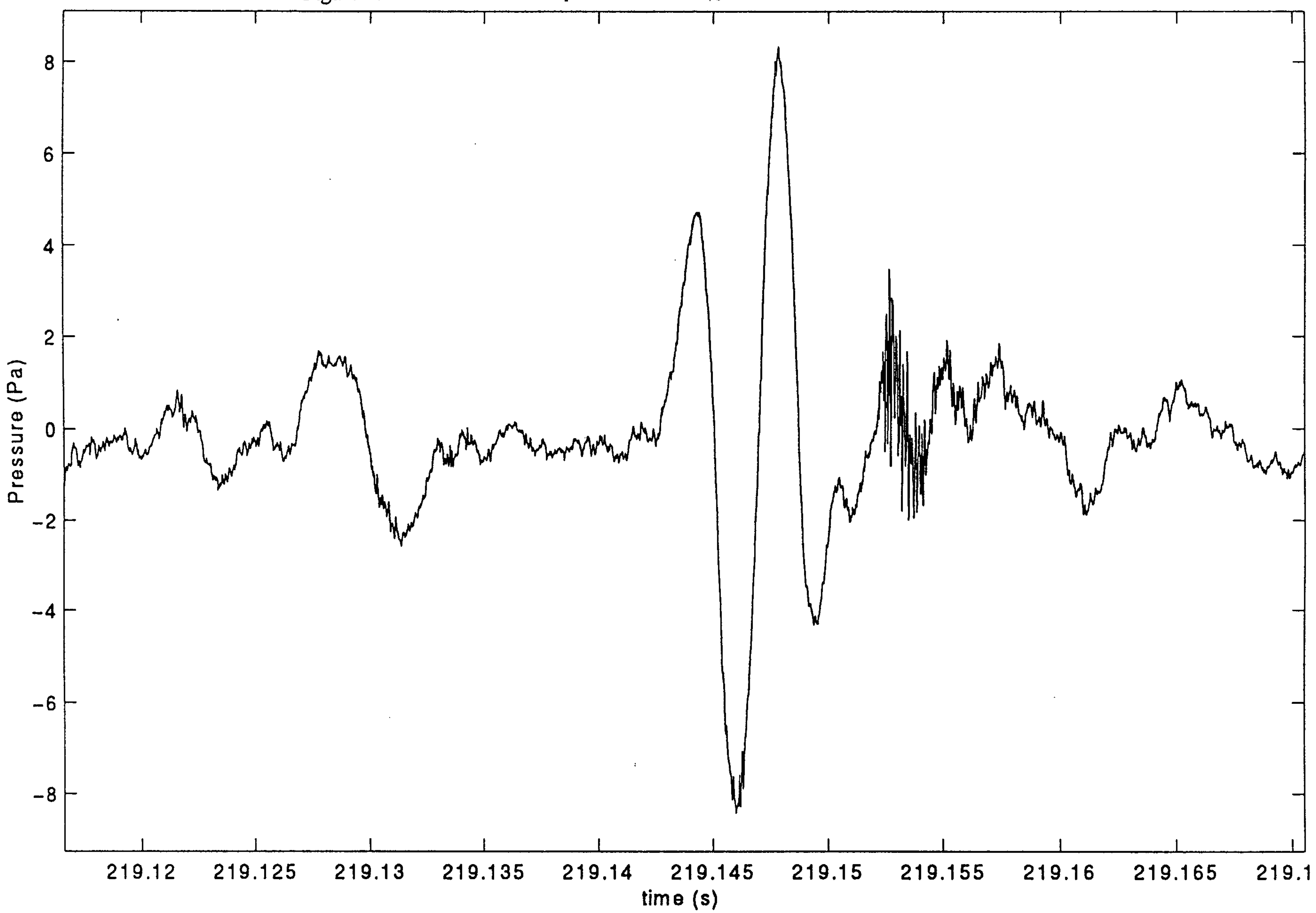


Figure 4: Raw time series sample from l6r1280.pcm large scale

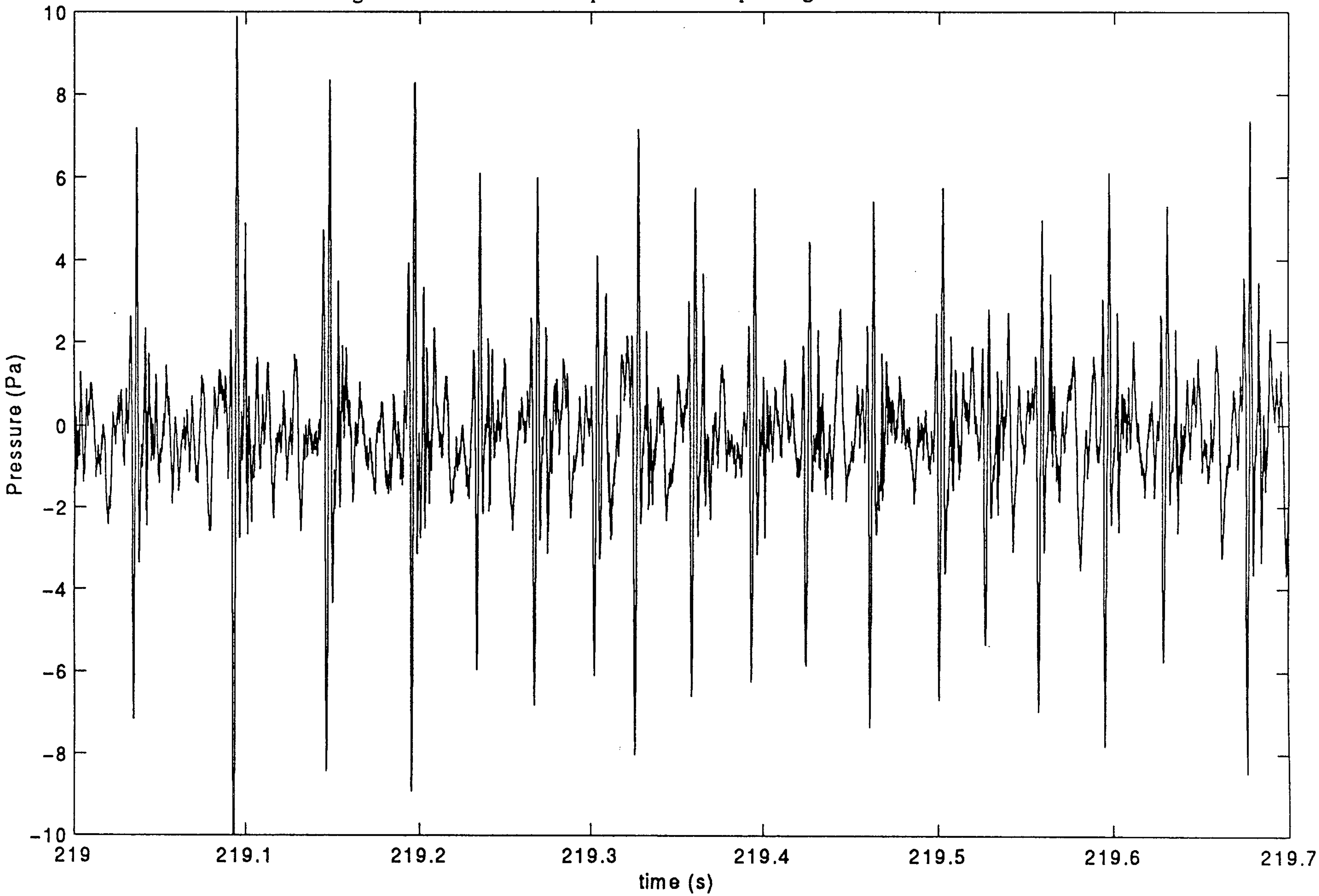


Figure 5: Pulse period versus height for l6r1280.pcm from 215-230 s

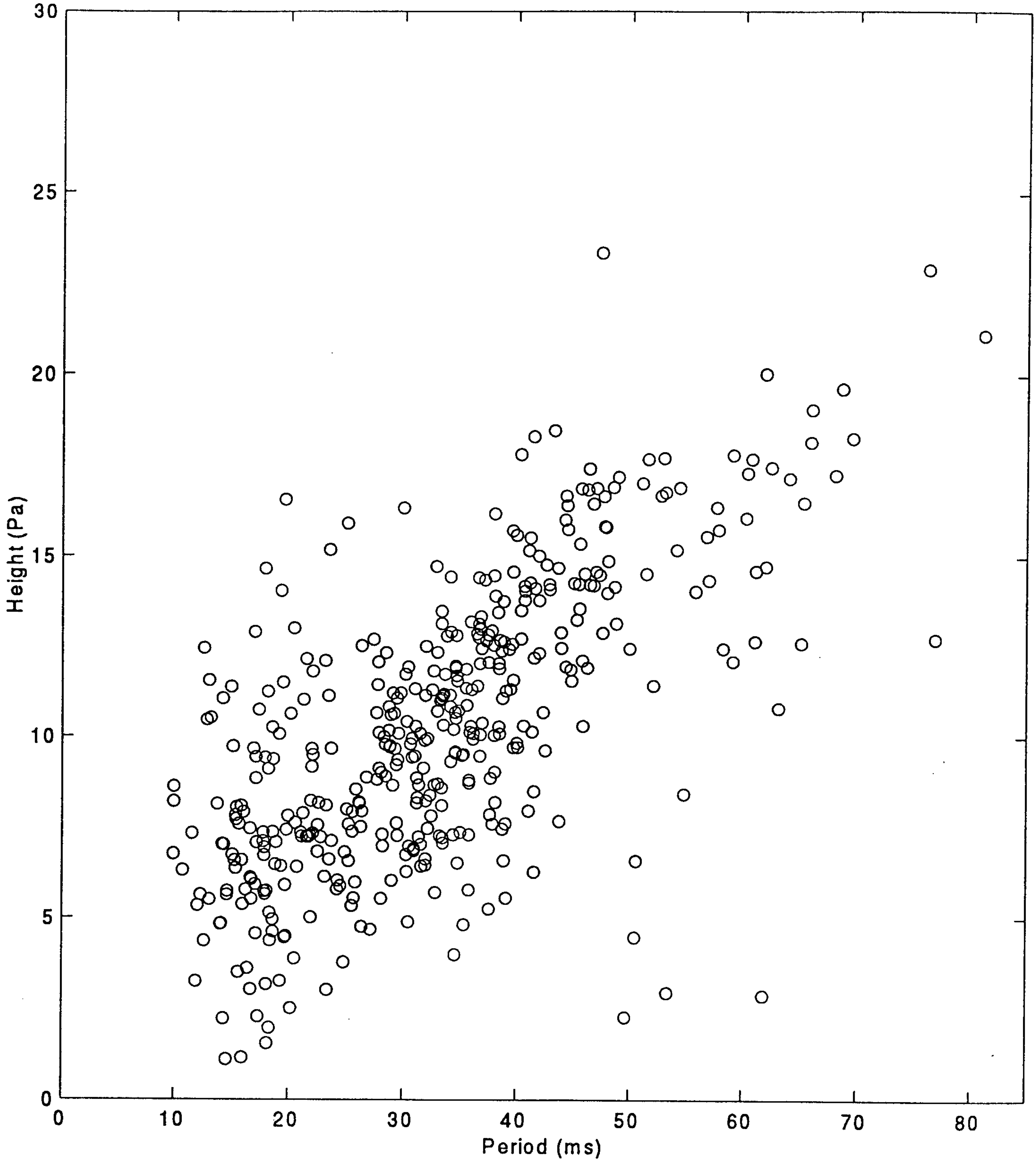


Figure 6: Raw time series from l9r4291.pcm

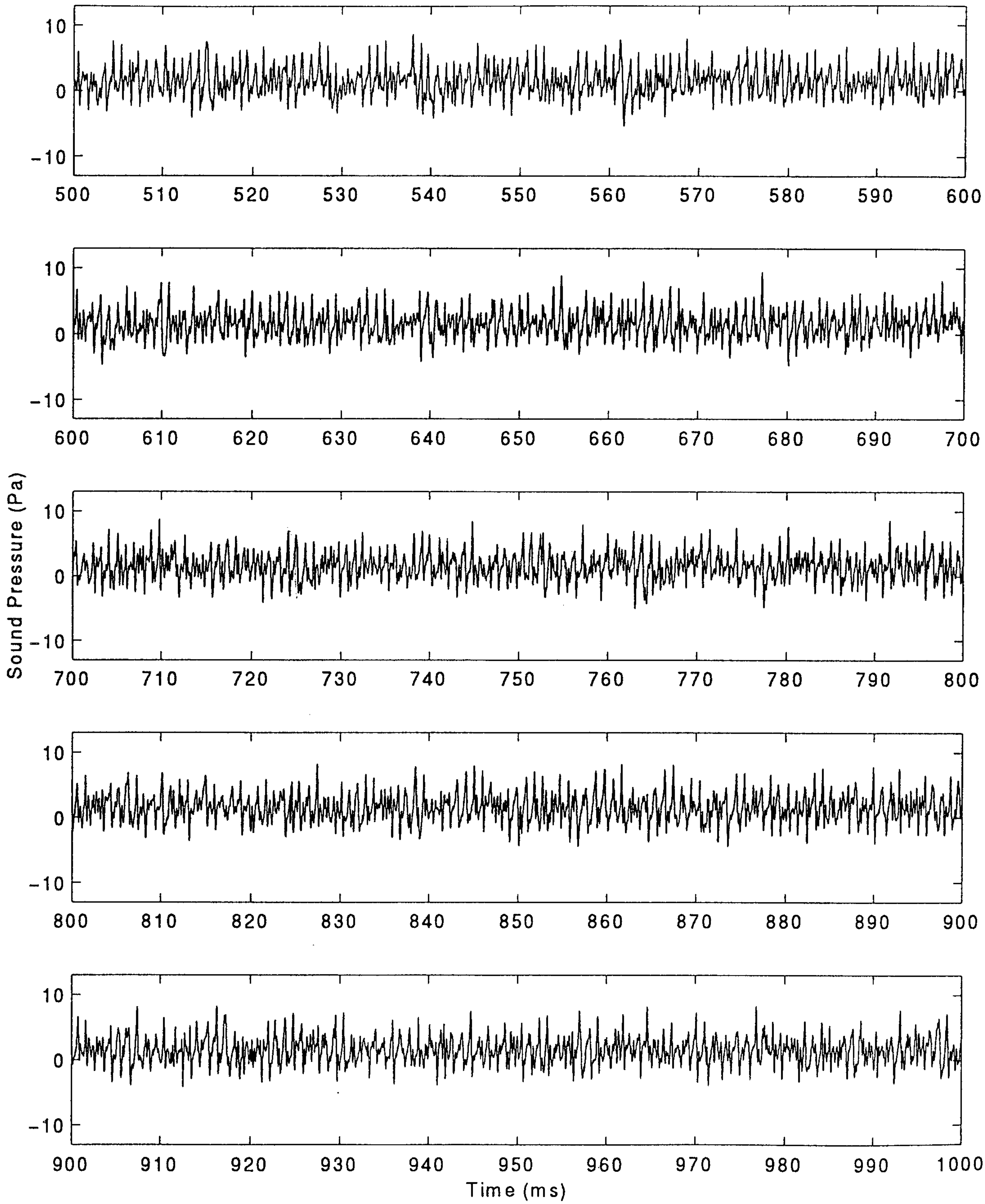


Figure 7: Autocorrelated signal of l9r4291.pcm made with meshplot.m

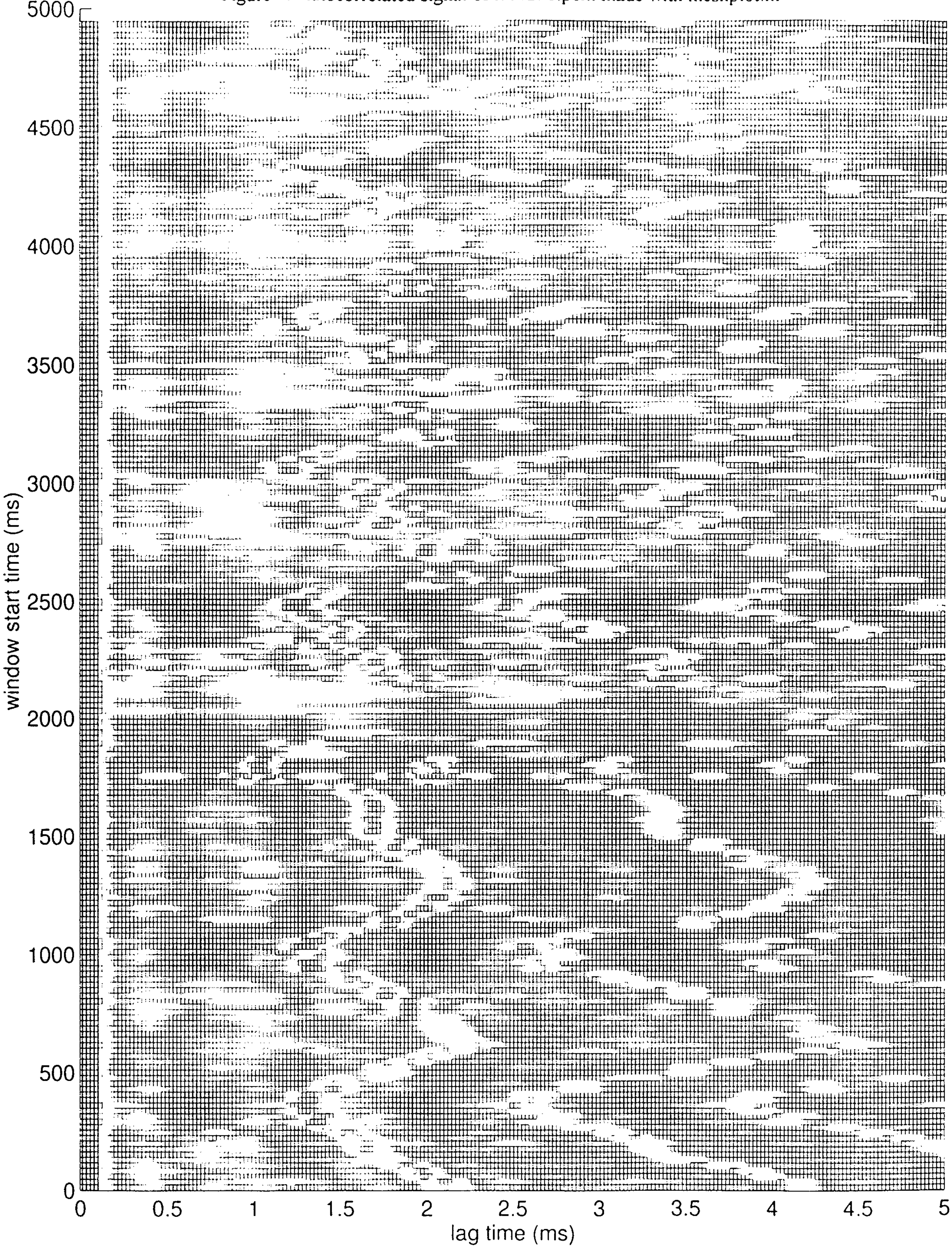


Figure 8: Time series of period and height from l9r4291.pcm

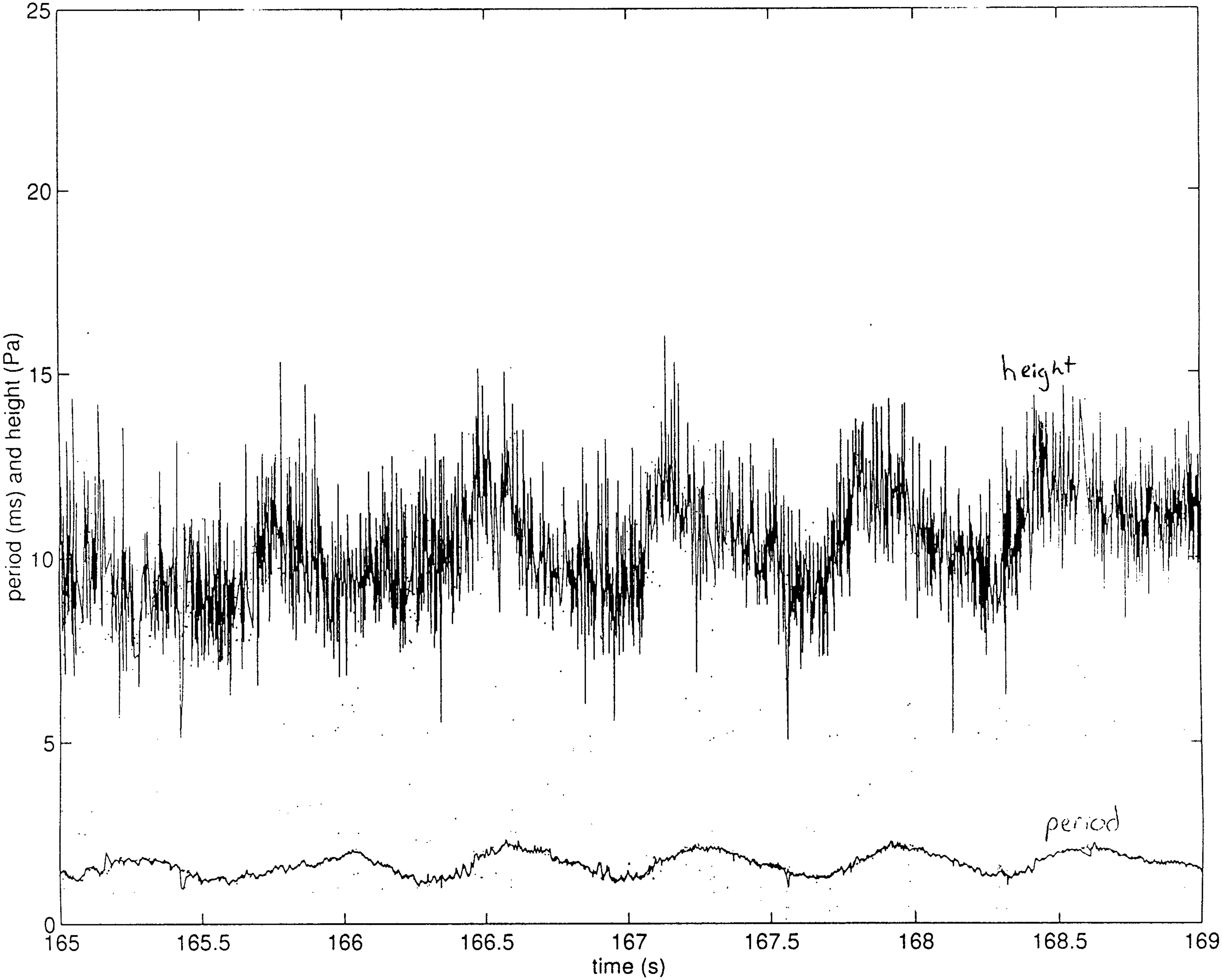


Figure 9: Conditionally sampled and averaged data from l6r1280.pcm

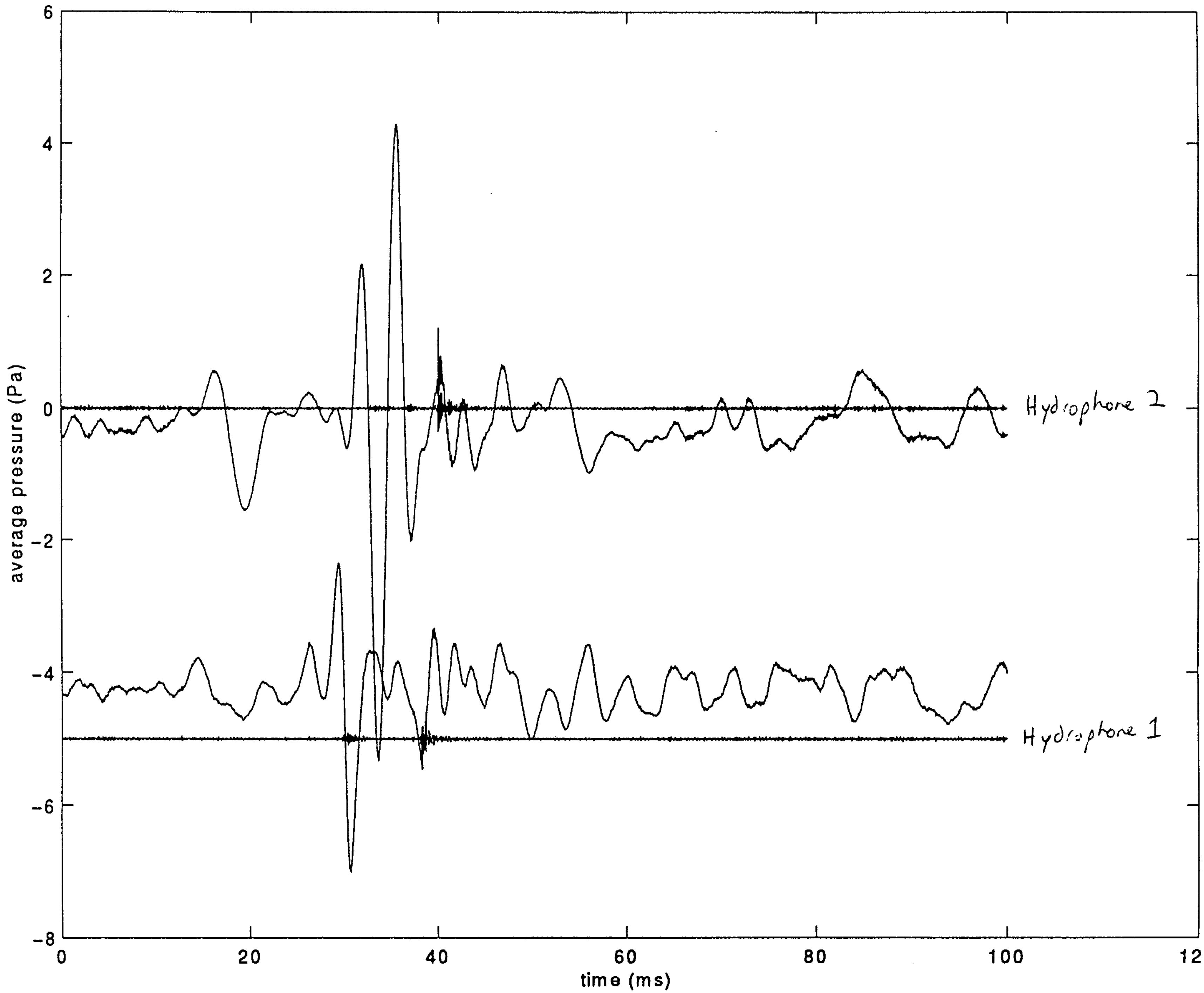


Figure 10: Average power spectral density of 2 ms following high frequency trigger

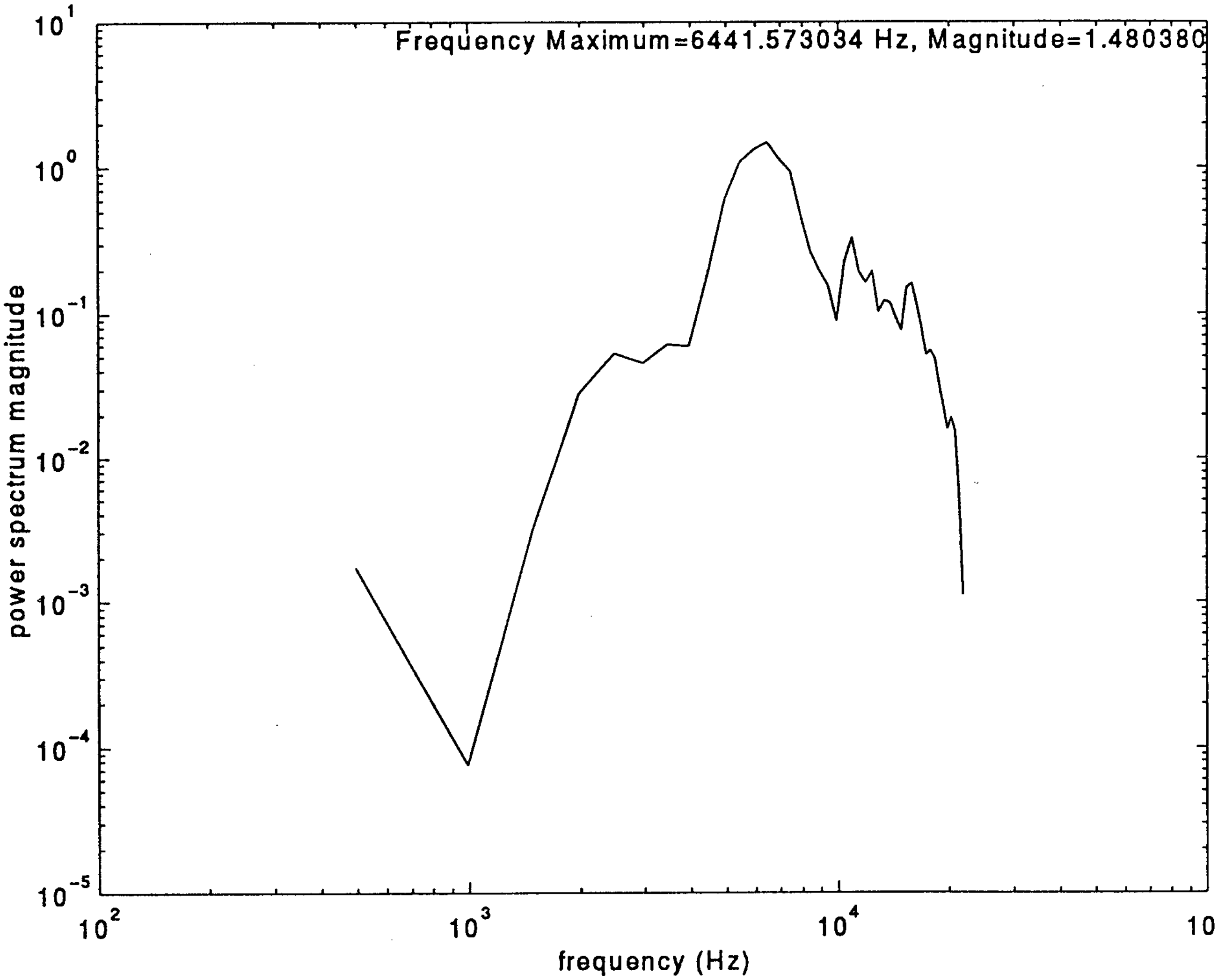


Figure 11: Conditionally sampled high frequency acoustic signal exponentially fitted

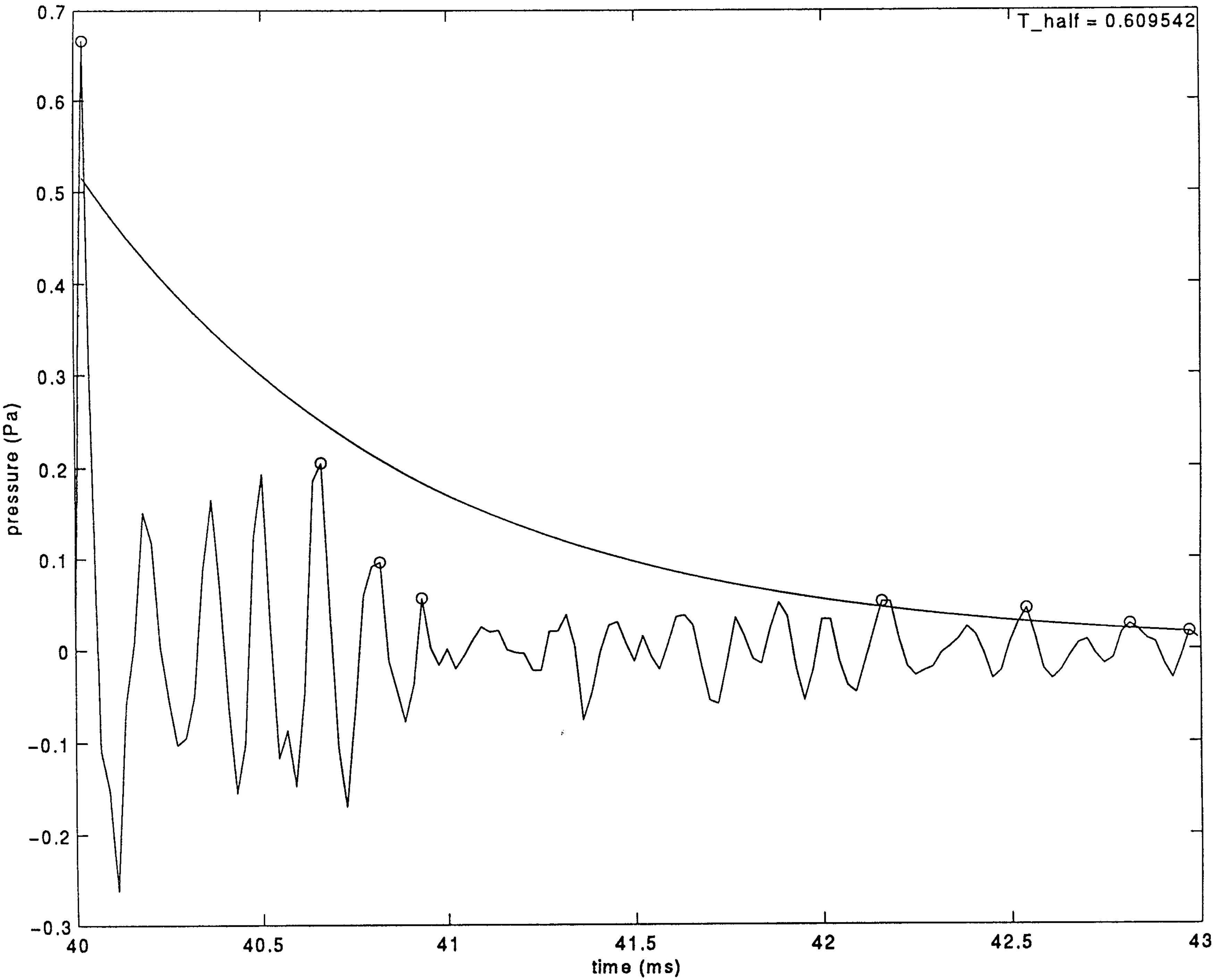


Figure 12: Signal paramater plots produced from l6r1280.pcm by param.m

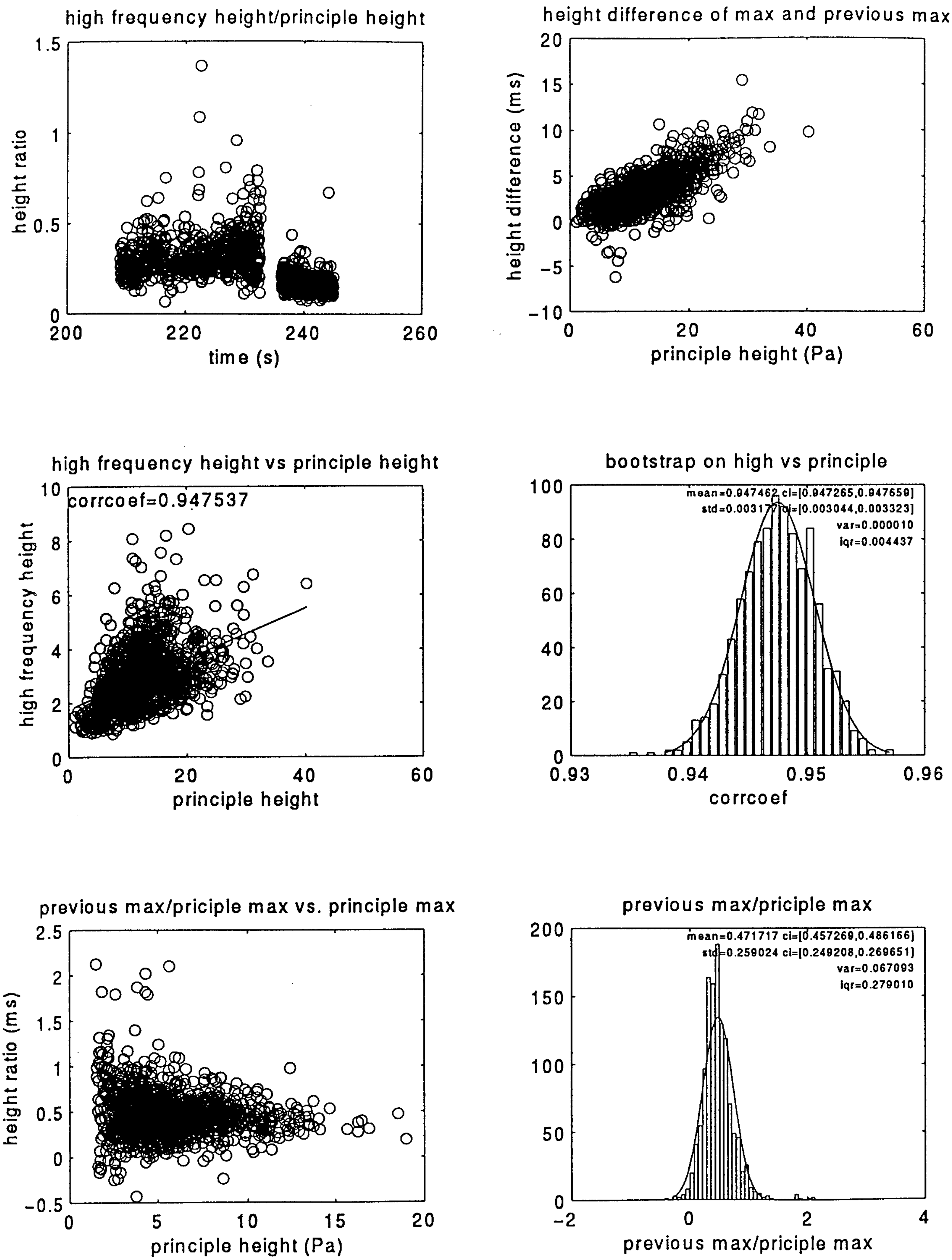
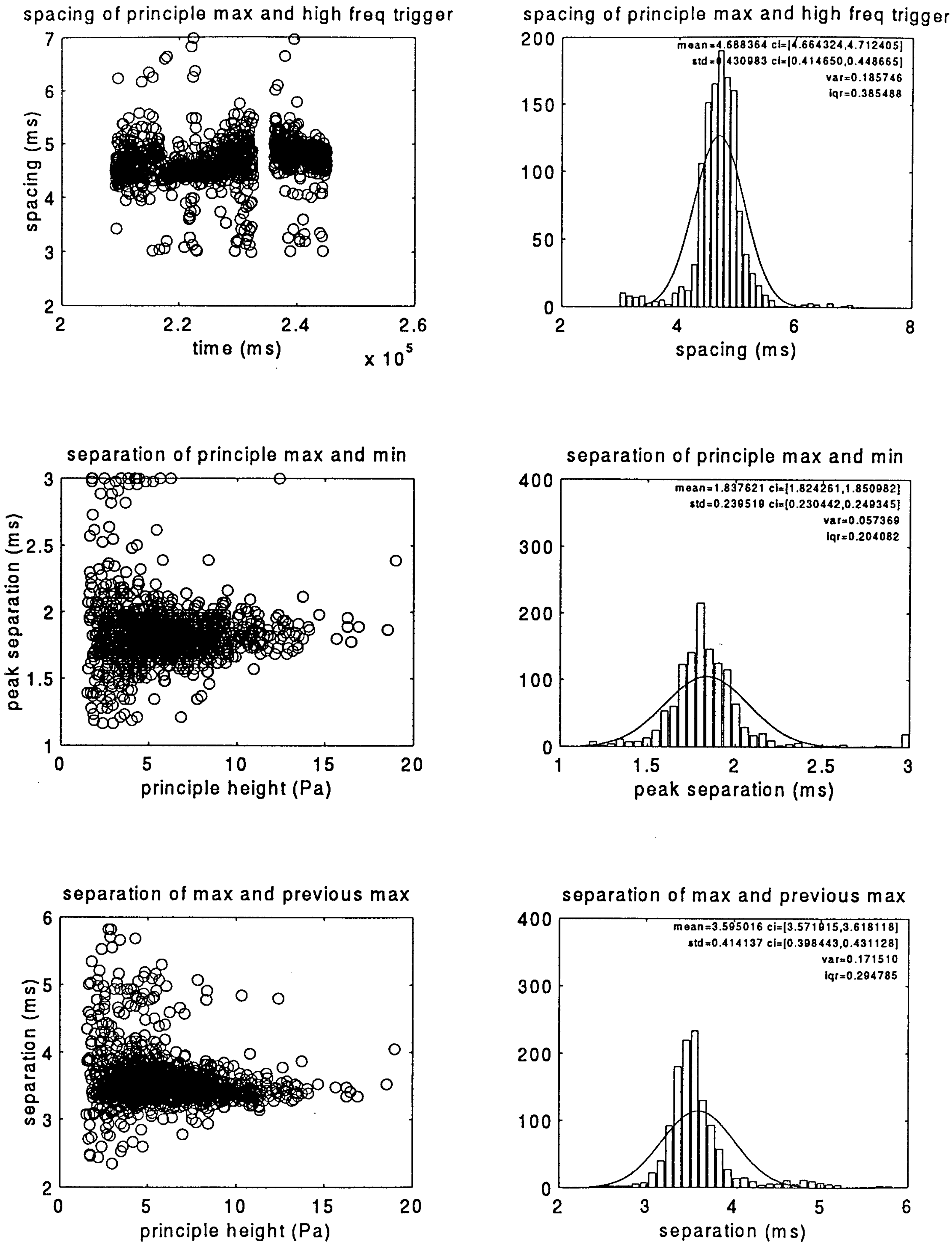


Figure 13: Signal paramater plots produced from l6r1280.pcm by param.m



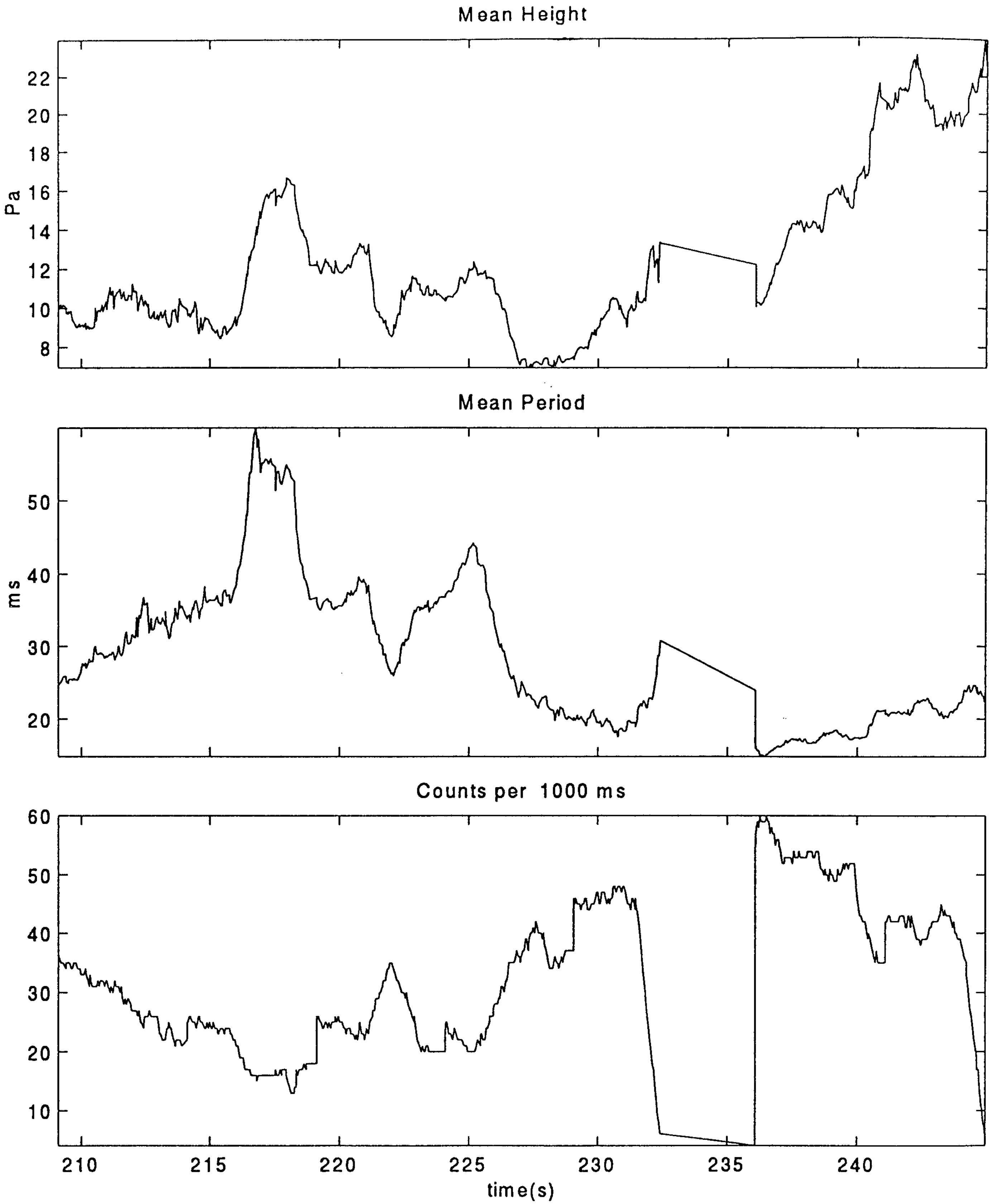



Figure 14: Time series of parameters from l6r1280.pcm produced by periodn.m

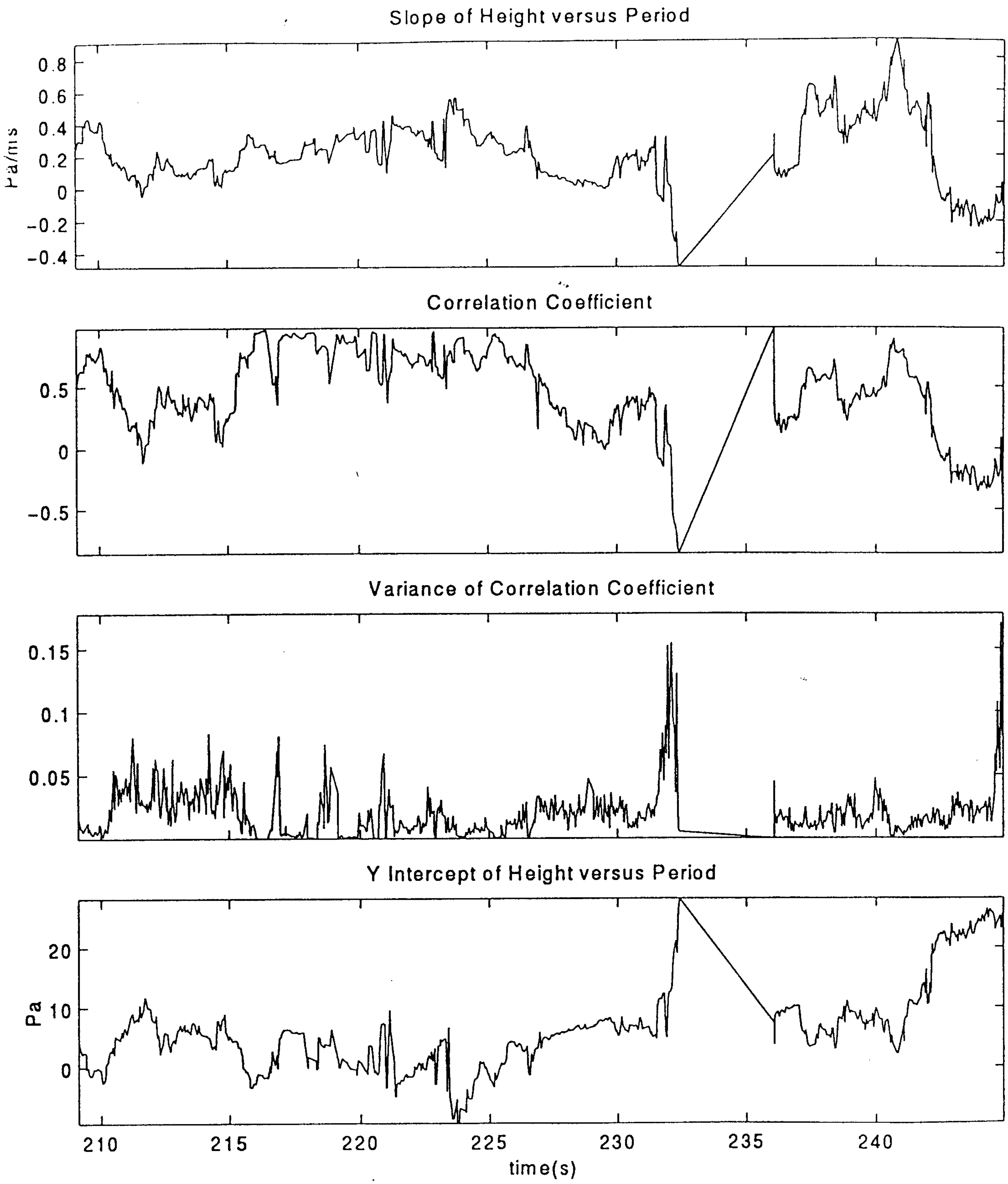



Figure 15: Time series of height versus period linear relationship parameters produced by period.m

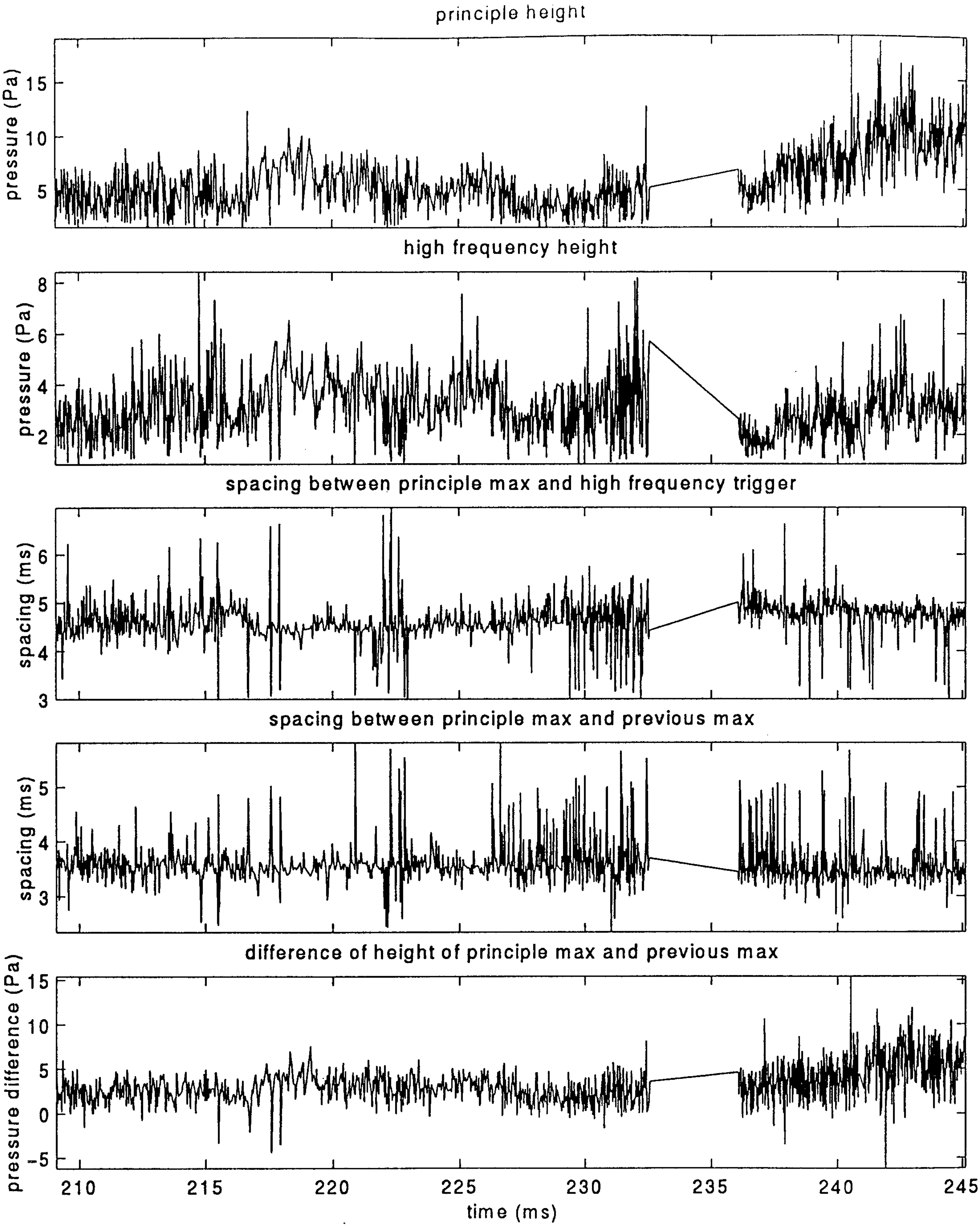



Figure 16: Time series of signal parmeters from l6r1280.pcm produced by series.m

Figure 17: Spectral information of acoustic wave from l6r1280.pcm produced by spectral.m

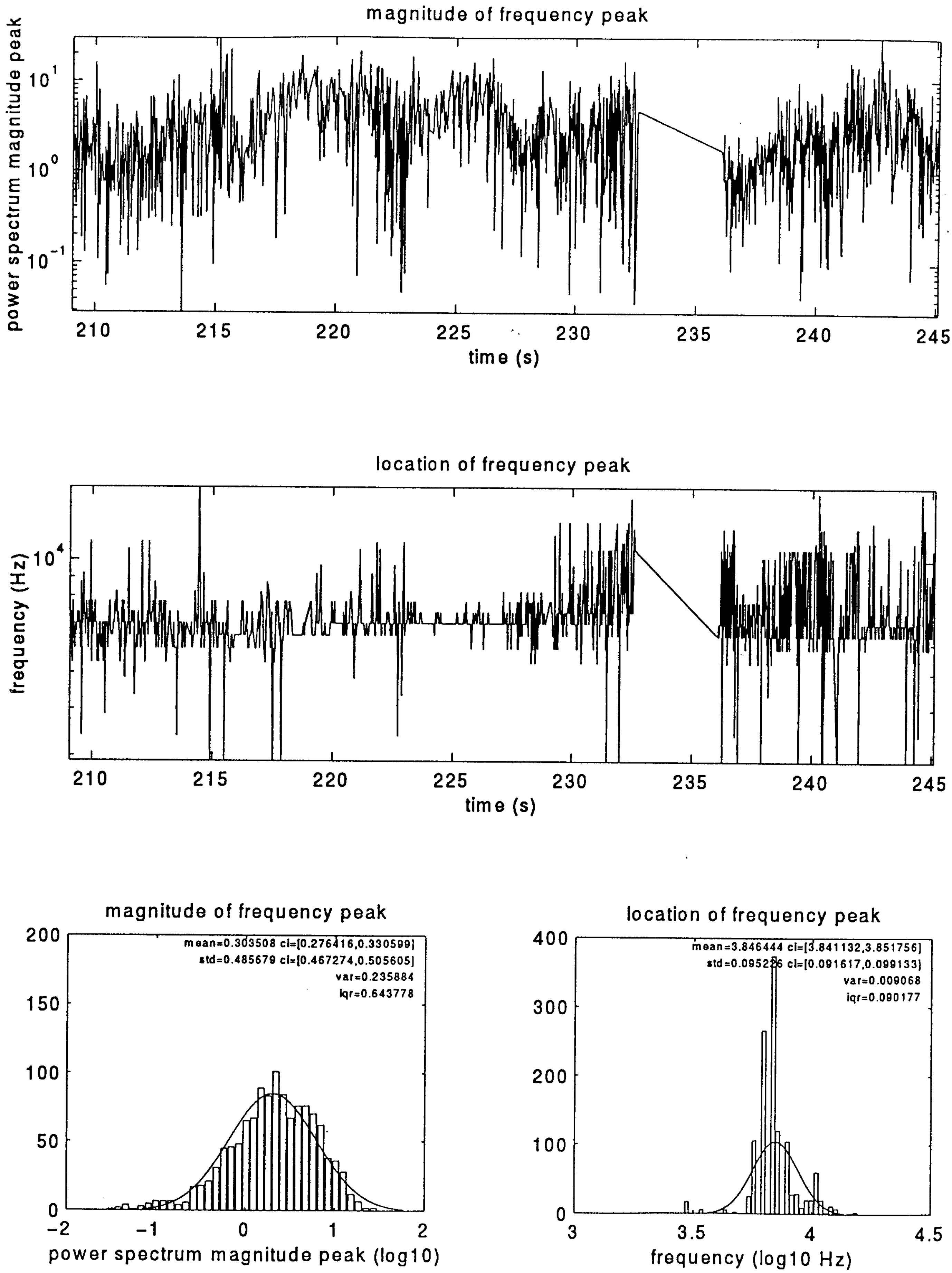


Figure 18: Time series of period and height from l9r4291.pcm

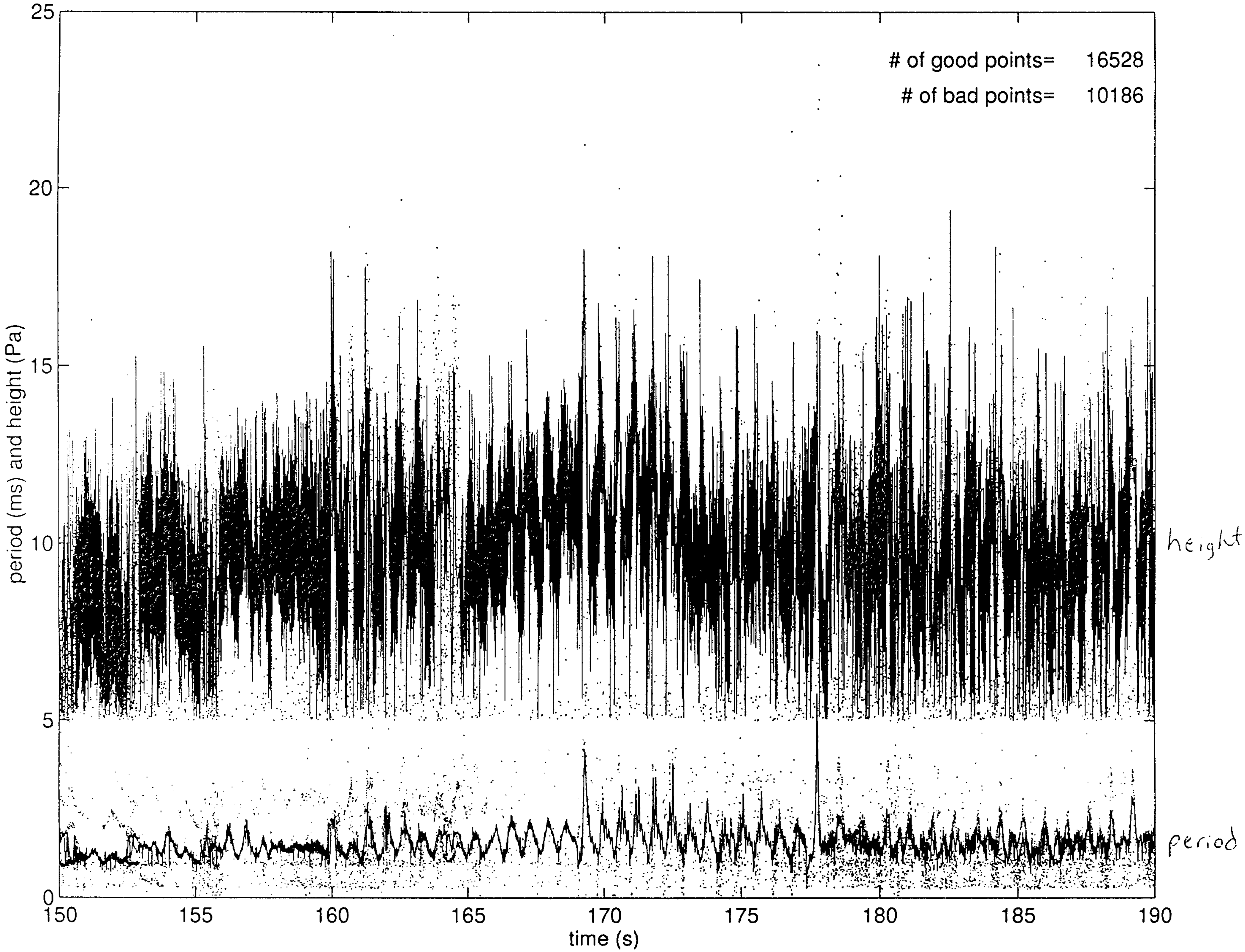
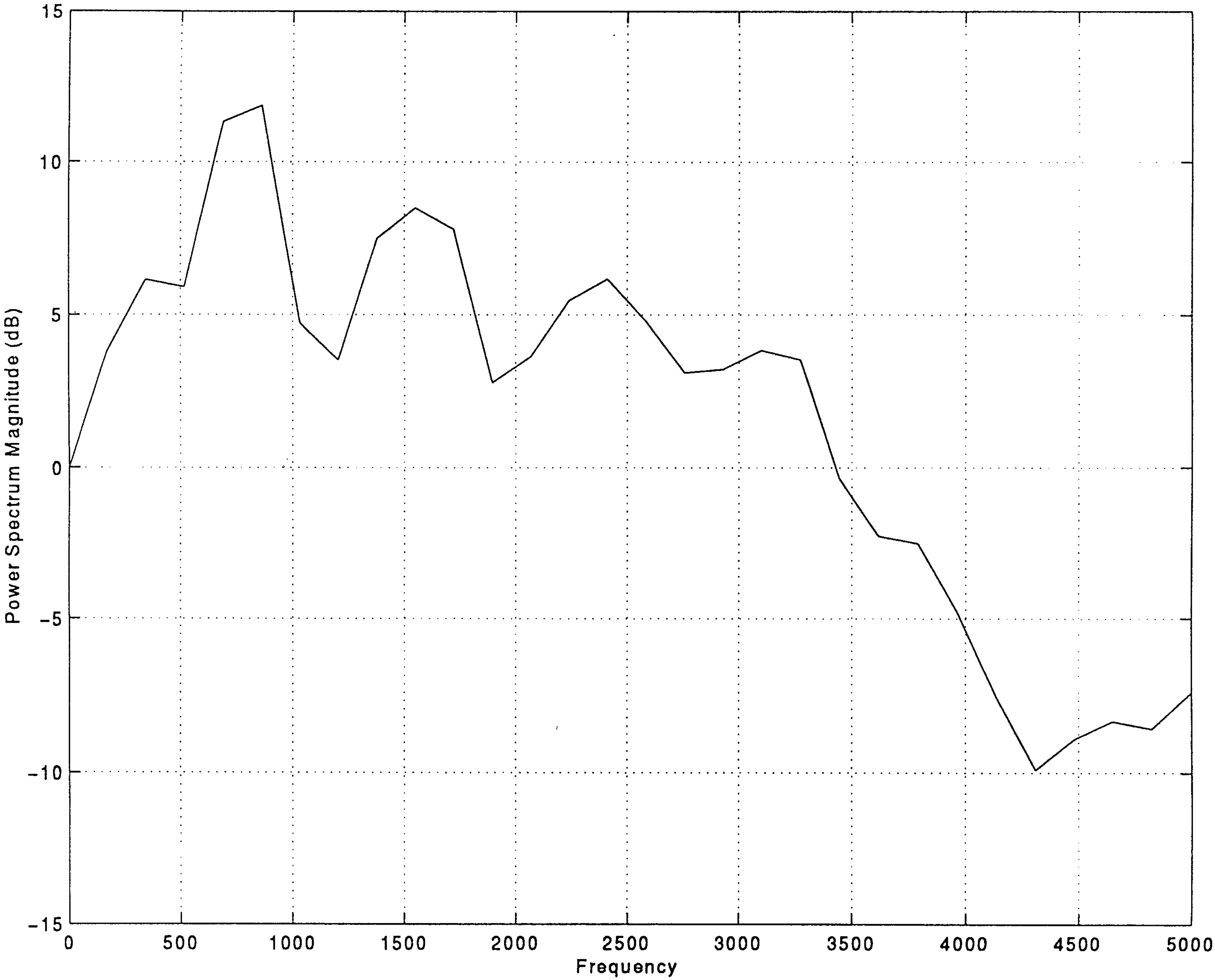


Figure 19: Power spectrum of l9r4291.pcm from 165-166 s



## Acknowledgements

I appreciate all the suggestions and guidance from both Dr. David Farmer and Dr. Yunbo Xie. Grace Kamitakahara-King was very helpful with computer questions and problems.

## References

Xie, Y. and D. M. Farmer, "Acoustic and Seismeic Perspective of Ice Events Observed in SIMI '94 Experiment." December 1994, unpublished.

## Appendix One: read2.m

The function read2 reads pcm files logged with the logging program on OMOI called E:\YUNBO\log93.exe. When logged the data is stored in a file G:\YXIE_DAT\temp.pcm on OMOI. This output file name is hardwired into log93.exe. After creation of temp.pcm the name needs to be changed to avoid overwriting.

```
function a=read2(file,start,len)
%READ2: reads an interleaved two channel data stream of 2 byte integers
%     (eg. the anita93 format of pcm files)
%
%     OUT=READ(FILE,START,LEN) gives the matrix OUT. FILE is a string in
%     'quotation marks', START is a time in ms from the beginning of the file
%     to start reading, and LEN is the length of time in ms to read after file.
%     The first column of OUT is the time in ms relative to the start of the
%     data file. The second column is the data from channel 1 and second column
%     is the data from channel 2. It begins at time START (in ms) from the start
%     of the data file and takes LEN ms of data following START.
format long e
freadid=fopen(file,'r');
status=fseek(freadid,2*2*44.1*start,'bof');        % first 2 is for 2 byte integers
                                                   % second 2 is for 2 channels
                                                   % 44.1 is for sample rate in kHz
                                                   % start is for amount to index in ms

if status==-1
 error('status=-1 on fseek')
end

F=fread(freadid,1,'integer*2');
if(rem(F,2)==0)             %F is even and therefore from channel 1
        status=fseek(freadid,2*2*44.1*start,'bof');

        if status==-1
                error('status=-1 on fseek')
        end
end

F=fread(freadid,2*44.1*len,'integer*2');           % 2 is for 2 channels
                                                   % 44.1 is for sample rate in kHz
                                                   % len is for amount to index in ms

status=fclose(freadid);
if status==-1
 error('status=-1 on fclose')
end

lenF=length(F);
if lenF < 2*44.1*len
        len=floor(lenF/(2.0*44.1));
end
chan=1;

a=[(1/44.1+start):(1/44.1):(len+start);1.152e-3*F(chan:2:(2*len*44.1))';1.152e-3*F(chan+1:2:(2*len*44.1))']';
```

# Appendix Two: double.m

The following matlab function is for the analysis of l6r1280.pcm type data.

A complete example of running double.m is

```
matlab>>a=read2('g:\l6r1280.pcm",3.4*60*1000+5*1000,5*1000);
matlab>>double([a(:,1) a(:,3) a(:,2)], 1.5, 0.5, 10, 3, 7, 'a');
matlab>>plots %after editing the plots.m file to read triggera.dat
matlab>>series % after editing the series.m file to read triggera.dat
matlab>>prama % after editing the param.m file to read triggera.dat
matlab>>spectral % after editing the spectral.m file to read triggera.dat
```

```
function double(a, th1, th2, win, gap, highgap, letter)

% Double was written to extract the signal parameters of the l6r1280.pcm type data.
%
% Input Parameters
%
% Double takes a three column raw data vector called 'a'. The first column is time,
% the second column is raw data from whichever hydrophone you wish to trigger on,
% and the third column is the other raw data signal.
%
% th1 is the large threshold (in Pascals) intended to be broken by the SH-wave
% th2 is the second threshold (in Pascals) to be broken by the high pass filtered data
% win is the length of time (in ms) after the maximum value of the SH wave to start
%  looking for the next SH-wave
% gap is the length of time after the maximum value of the SH wave to begin looking
%  for a highfrequence trigger in the acoustic wave
% highgap is the the length of time after the maximum value of the SH wave to stop
%  looking for a high frequence trigger in the acoustic wave
% letter is just a letter to append to the files created here to avoid overwriting
%  from previous runs
%
% Three files are created containing signal parameters to use when creating plots
%  d:\anita94b\asciidat\trigger*.dat contains the time and heights of certain points
%   the raw data
%  d:\anita94b\asciidat\frequen*.dat contains the averaged spectrum of the filtered
%   acoustic signal
%  d:\anita94b\asciidat\sig*.dat contains the conditionally sampled and averaged data
%   for time of arrival determination
%
% Several plots are also made
%  conditionally sampled data
%  spectrum of filtered acoustic wave
%  the fitted curve of the acoustic wave decay

format long e
S=['d:\anita94b\asciidat\trigger' letter '.dat'];
trigid=fopen(S,'w');
S=['d:\anita94b\asciidat\frequen' letter '.dat'];
freqid=fopen(S,'w');

[d,c]=butter(7,1000/22050,'high');
[h,g]=butter(7,2000/22050,'high');
win=win*44.1;
gap=gap*44.1;        %minimum gap between principle trigger and high freq signal
highgap=highgap*44.1; %end of window for high frequency

k=0;
```

```
%i=max(2*win+1,41*44.1);

i=44.1*40+1;

len=length(i-44.1*40:i+60*44.1);
signal=zeros(len,3);
signal(:,1)=[1/44.1:1.0/44.1:len/44.1]';

b=[a(:,1) filtfilt(h,g,a(:,2))];
w=[a(:,1) filtfilt(d,c,a(:,2))];
%a(1,1)
%w(1,1)

%temp=max(len,4*win);
temp=60*44.1+1;
while i <= (length(a)-2*temp)

  if a(i,2) >= th1
     [G,L]=max(a(i:i+win,2));       %find the max after the trigger
     i=i+L-1;                     %place index at the max
     y=a(i-win:i+win,:);           %grab a chunk around max

     z=w(i+gap:i+highgap,:);                  %grab a filtered chunk after max for high freq search
     maximum=max(z(:,2));         %find the max in z
     if maximum >= th2            %a good signal has been found
        k=k+1;               %increment the # of signals found
        principlemaxtime = a(i,1);        %time of principle max
        principlemaxheight=a(i,2);        %height of principle max

        for j=1:length(z)                %find the time where th2 is broken
           if z(j,2) >= th2
                hightriggertime=z(j,1);
                break;
           end % th2
        end %for j

        %get stress release time here
        segmentlength=3*44.1;
        %hightriggertime
        %w(1,1)
        tempor=(hightriggertime-w(1,1))*44.1;
        [A,lambda]=decay([w(tempor:tempor+segmentlength,1)-hightriggertime
w(tempor:tempor+segmentlength,2)],0);
        Thalf=real(6.931471805599453e-001/lambda);

        J=length(y)/2.0;

        [N,K]=min(y(J-3*44.1:J,2));             %previous min
        K=K+J-3*44.1-1;
        [O,L]=max( y(K-3*44.1:K,2));             %previous max
        L=L+K-3*44.1-1;
        principleheight = G - N;          %max height of signal
        deltamax = G - O;                 %difference in height of max and previous max
        separation = (J-K)/44.1;           %time separation of principle max and previous min
        separation2 = (J-L)/44.1;          %time separation of principle max and previous max
        highheight = max( z(:,2) ) - min( z(:,2) ); %height of high frequency signal

        ss = z(j,1)-a(i,1);  %time spacing between principle trigger and high frequency trigger
```

```
        fprintf(trigid,'%20.19e\t%20.19e\t%20.19e\t%20.19e\t%20.19e\t%20.19e\t%20.19e\t%20.19e\t%20.
19e\t%20.19e\n', principlemaxtime, principlemaxheight, hightriggertime, principleheight, highheight, ss,
separation, separation2, deltamax, Thalf);

% summed signal
        lineup=i+gap+(j-1);
        signal(:,2)=signal(:,2)+a(lineup-44.1*40:lineup+60*44.1,2);
        signal(:,3)=signal(:,3)+a(lineup-44.1*40:lineup+60*44.1,3);


% filtered high frequency signal spectrum
        [Pxx(:,2),Pxx(:,1)] = psd(b(i+gap+j:i+gap+j+2*44.1,2),[],44100);

        inc=length(Pxx(:,1))/max(Pxx(:,1));
        [peak,index]=max(Pxx(:,2));
        index=index/inc;
        fprintf(freqid,'%20.19e\t%20.19e\t%20.19e\n',hightriggertime,peak,index);
        if k==1
            fre=Pxx(:,1);
            PXX=zeros(length(Pxx),1);
        end
        PXX=PXX+Pxx(:,2);

        i=i+win;
      end %if th2
   end %if th1

   i=i+1;

end %while
k
fclose(trigid);
fclose(freqid);

%clear principlemaxtime principlemaxheight hightriggertime principleheight highheight ss separation
separation2 deltamax peak index

signal(:,2)=signal(:,2)/(k);    %average the signals
signal(:,3)=signal(:,3)/(k);
PXX=PXX/(k);

figure
plot (signal(:,1),signal(:,2),'w',signal(:,1),signal(:,3)-4,'r')
title('Average of signals from 1334')
ylabel('average pressure (Pa)')
xlabel('time (ms)')
orient landscape
hold on
filteredsig=filtfilt(h,g,signal(:,2));
plot (signal(:,1),filteredsig,'w',signal(:,1),filtfilt(h,g,signal(:,3))-5,'r')
hold off

S=['d:\anita94b\asciidat\sig' letter '.dat'];
sigid=fopen(S,'w');
for i=1:length(signal)
      fprintf(sigid,'%20.19e\t%20.19e\t%20.19e\n',signal(i,1),signal(i,2),signal(i,3));
end %for i
fclose(sigid)

segmentlength=3*44.1;
```

```
sig=[signal(40*44.1:40*44.1+segmentlength,1) filteredsig(40*44.1:40*44.1+segmentlength)];
[A,lambda]=decay(sig,1);

%--------------------------------------------------------------------------------
figure
loglog (fre,abs(PXX));
S=sprintf('average power spectral density of 2 ms following high frequency trigger (%4.0f signals)',k);
title(S)
ylabel('power spectrum magnitude')
xlabel('frequency (Hz)')

[Max,Index]=max(abs(PXX));

v=axis;
axis(v);
v=axis;
S=sprintf('Frequency Maximum=%f Hz, Magnitude=%f',fre(Index),Max);
h=text(v(2),v(4),S);
set(h,'HorizontalAlignment','right')
set(h,'VerticalAlignment','top')
```

## Appendix Three: finder.m

The following matlab function is for the analysis of 19r4291.pcm type data.

A complete example of running finder.m is

```
matlab>>finder('e:\anita94b\pcmdat\194291.pcm', 0, 10*1000, 1, 5);
matlab>>convert;
matlab>>despiken(period,0.4,0.4);
```

```
function finder(file, start, len, chan, heightthresh)
%
% FINDER performs a search for pressure drops in a time series of pressure.
%
% Input Parameters
%  File is string containing the name of the file where the raw time series resides.
%  Start is the position, in ms, from the start of the file you wish to process.
%  Len is the length of the file, in ms, after start that you wish to process.
%  Chan is the channel you wish to process.
%  Heightthresh, in Pa, is the pressure difference of a max/min pair that must be
%   exceeded for a pair to be called a pulse.
%
% Output file
%  A file called 'd:\anita94b\asciidat\long.dat' is produced. Its first four
%  columns are the time of the max and its pressure, the time of the min and
%  its pressure.



format long e
chunksize=100;    %size of timeseries to read in ms should be 10 or greater/ good for len to be a multiple
of chunksize
pntr=start;       %in ms

fid=fopen('d:\anita94b\asciidat\long.dat','w');

while (pntr < start+len)
        a=read(file,pntr,chunksize,chan);
        pntr=pntr+chunksize;
        lengtha=length(a);
        % search a to find max/min pairs
        i=3;
        while i <=lengtha
           if (a(i,2)<a(i-1,2) & a(i-2,2)<a(i-1,2))
          MAXI=i-1;
          while i <= lengtha %search for matching minimum
              if (a(i,2)>a(i-1,2) & a(i-2,2)>a(i-1,2)) %minimum => a pair has been found
                     up=a(MAXI,1);
                     MAX=a(MAXI,2);
                     down=a(i-1,1);
                     MIN=a(i-1,2);
                     MINI=i-1;
                     if ((MAX-MIN) > heightthresh)

        fprintf(fid,'%20.19e\t%20.19e\t%20.19e\t%20.19e\t%20.19e\t%20.19e\t%20.19e\n',a(MAXI,1), MAX,
a(MINI,1), MIN, up, down, (MAX-MIN)-heightthresh);
                     end
                     break;
              end %if
              i=i+1;
          end %while
```

```
            i=i+1;
            end %if
            i=i+1;
        end %while
end %while

fclose(fid)

%t=clock;info=findern('remote19',0,1000,1,b,175,8);etime(clock,t)

%plot the trigering info
%a=read('remote19',0,1000,1);
%figure;plot(a(:,1),a(:,2),'w',info(:,1),info(:,2),'ro',info(:,3),info(:,4),'go');

%plot principle height vs period
%height=info(:,2)-info(:,4);
%figure;plot(info(2:length(info),1)-info(1:length(info)-1,1),height(2:length(height)),'co');

%plot time series of principle height and period
%figure;plot(info(2:length(info),1),height(2:length(height)),'c',info(2:length(info),1),info(2:length(info),1)-
info(1:length(info)-1,1),'r');

%info=short;

%prepare vector for despike
%height=info(:,2)-info(:,4);
%height=height(2:length(height));
%period=info(2:length(info),1)-info(1:length(info)-1,1);
%period=[info(2:length(info),1),period,height];
```

## Appendix Four: despike.m

The following matlab function removes spikes from a time series of peirod.

```
function despike(input,filtdeviation,jumpdeviation)
%
% Despike removes spikes in the data in the second column of input.
% The first column is usually time, the second period and the third height.
%
% Input Parameters
%  input is a three column vector: first column time,
%   second column for despiking, and the third column is just a tag along column
%  filtdeviation is the cut off deviation for the difference
%   between the second column value and the filtered second column value
%  jumpdeviation is the cut off deviation for a point in column two to the
%   preceeding point in column two
%
% Output Files
%  Two files are produced
%   'd:\anita94b\asciidat\good.dat' contains the good points of input
%   'd:\anita94b\asciidat\bad.dat' contains the spikes of input
%
% Plot
%  The time series of the second and thrid colums are ploted. The good points are
%  ploted linked together and the bad points are ploted as dots

format long e
fidgood=fopen('d:\anita94b\asciidat\good.dat','w');
fidbad=fopen('d:\anita94b\asciidat\bad.dat','w');

[d,c]=butter(7,2000/22050);
per=filtfilt(d,c,input(:,2));

%g=1;
%b=0;
last=input(1,2);
%good(g,:)=input(g,:);
fprintf(fidgood,'%20.19e\t%20.19e\t%20.19e\n',input(1,1),input(1,2),input(1,3));

for i=2:length(input)
        if (abs(input(i,2)-per(i))<filtdeviation | abs(input(i,2)-last)<jumpdeviation)
          %g=g+1;
          fprintf(fidgood,'%20.19e\t%20.19e\t%20.19e\n',input(i,1),input(i,2),input(i,3));
          %good(g,:)=input(i,:);
          last=input(i,2);
        else
          fprintf(fidbad,'%20.19e\t%20.19e\t%20.19e\n',input(i,1),input(i,2),input(i,3));
          %b=b+1;
          %bad(b,:)=input(i,:);
        end %if else
end %for

fclose(fidgood);
fclose(fidbad);

load d:\anita94b\asciidat\good.dat
load d:\anita94b\asciidat\bad.dat

figure
plot(good(:,1)/1000,good(:,2),'g');
```

```
hold on
plot(good(:,1)/1000,good(:,3),'c');
plot(bad(:,1)/1000,bad(:,2),'r.','markersize',2);
h=plot(bad(:,1)/1000,bad(:,3),'.w','markersize',2);
S=sprintf('despike(period,%3.2f,%3.2f)',filtdeviation,jumpdeviation);
title(S);
xlabel('time (s)')
ylabel('period (ms) and height (Pa)')
orient landscape;

v=axis;
axis([v(1) v(2) 0 25])
v=axis;
axis(v);
S=sprintf('# of good points=%10.0f',length(good));
h=text(v(2)-0.4,v(4)-1,S);
set(h,'HorizontalAlignment','right')
set(h,'VerticalAlignment','top')
S=sprintf('# of bad points=%10.0f',length(bad));
h=text(v(2)-0.4,v(4)-1-1*1,S);
set(h,'HorizontalAlignment','right')
set(h,'VerticalAlignment','top')

h=text(v(1)+0.4,v(4)-1,'height');
set(h,'HorizontalAlignment','left')
set(h,'VerticalAlignment','top')
set(h,'Color','c');
h=text(v(1)+0.4,v(4)-1-1*1,'period');
set(h,'HorizontalAlignment','left')
set(h,'VerticalAlignment','top')
set(h,'Color','g');
```